A Heuristic-Based Methodology for Semantic Augmentation of User Queries on the Web*

Andrew Burton-Jones[1], Veda C. Storey[1], Vijayan Sugumaran[2], and Sandeep Purao[3]

[1] J. Mack Robinson College of Business, Georgia State University,
Atlanta, GA 30302,
{vstorey,abjones}@gsu.edu

[2] School of Business Administration, Oakland University
Rochester, MI 48309
sugumara@oakland.edu

[3] School of Information Sciences & Technology, The Pennsylvania State University,
University Park, PA 16801-3857
spurao@ist.psu.edu

Abstract. As the World Wide Web continues to grow, so does the need for effective approaches to processing users' queries that retrieve the most relevant information. Most search engines provide the user with many web pages, but at varying levels of relevancy. The Semantic Web has been proposed to retrieve and use more semantic information from the web. However, the capture and processing of semantic information is a difficult task because of the well-known problems that machines have with processing semantics. This research proposes a heuristic-based methodology for building context aware web queries. The methodology expands a user's query to identify possible word senses and then makes the query more relevant by restricting it using relevant information from the WordNet lexicon and the DARPA DAML library of domain ontologies. The methodology is implemented in a prototype. Initial testing of the prototype and comparison to results obtained from Google show that this heuristic based approach to processing queries can provide more relevant results to users, especially when query terms are ambiguous and/or when the methodology's heuristics are invoked.

1 Introduction

It is increasingly difficult to retrieve relevant web pages for queries from the World Wide Web due to its rapid growth and lack of structure [31, 32]. In response, the Semantic Web has been proposed to extend the WWW by giving information well-defined meaning [3, 10]. The Semantic Web relies heavily on ontologies to provide taxonomies of domain specific terms and inference rules that serve as surrogates for semantics [3]. Berners-Lee et al. describe the Semantic Web as "not a new web but an extension of the current one, in which information is given well-defined meaning" [3]. Unfortunately, it is difficult to capture and represent meaning in machine-

* This research was partially supported by J. Mack Robinson College of Business, Georgia State University and Office of Research & Graduate Study, Oakland University.

I.-Y. Song et al. (Eds.): ER 2003, LNCS 2813, pp. 476–489, 2003.

readable form, even though understanding more of the semantics of a user's application or query would help process users' queries more effectively.

There is wide agreement that a critical mass of ontologies is needed for representing semantics on the Semantic Web [6, 27]. Libraries of ontologies are being developed for this purpose, with the most well-known being the DARPA DAML library with approximately 200 ontologies and over 25,000 classes http://www.daml.org/ontologies/. Although significant effort has gone into building these ontologies, there is little research on methodologies for retrieving information using them. Thus, the development of a methodology for doing so would greatly assist in realizing the full potential of the Semantic Web.

The objective of this research, therefore, is to: *develop a heuristic-based methodology for an intelligent agent to process queries on the Semantic Web so that the processing takes into account the semantics of the user's request*. This is done through the use of WordNet [15] (http://www.cogsci.princeton.edu/cgi-bin/webwn) to obtain senses for query terms and WordNet and the DAML ontologies to augment a query by expanding and shrinking the set of query terms to achieve a more precise, context-specific query.

The contribution of this research is to develop a methodology that more effectively processes queries by capturing and augmenting the semantics of a user's query. The methodology has been implemented in a prototype and its effectiveness verified. Results of this research should help realize the potential of the Semantic Web, while demonstrating useful applications of lexicons and ontology libraries.

2 Related Work

2.1 Semantic Web

The unstructured nature of the web makes it difficult to query and difficult for applications to use. The Semantic Web is a vision of the web in which these problems will be solved by 'marking-up' terms on web pages with links to online ontologies that provide a machine-readable definition of the terms and their relationship with other terms [3]. This is intended to make the web machine-readable and, thus, easier to process for both humans and their applications (e.g., agents). Consider a prototypical query (adapted from [3]): *Find Mom a specialist who can provide a series of bi-weekly physical therapy sessions*. To complete this query on the Semantic Web, agents will use online ontologies to interpret relevant semantics on web pages (e.g., specialist, series, etc) [16, 25]. The proliferation of ontologies is crucial for the Semantic Web, hence the creation of large ontology libraries [10, 47]. Hendler [25] predicts: "*The Semantic Web…will not primarily consist of neat ontologies...I envision a complex Web of semantics ruled by the same sort of anarchy that rules the rest of the Web*." Therefore, research is needed to determine how useful these ontologies will be for helping users and agents query the Semantic Web.

2.2 Ontologies

An ontology should be a way of describing one's world [50]. Ontologies generally consist of terms, their definitions, and axioms relating them [19]. However, there are many different definitions, descriptions, and types of ontologies [20, 39, 40]. In knowledge representation, well known contributions include Ontolingua [14], SHOE [23], Cyc [21] and the XML based schemes such as OIL [16], and DAML [24]. Ontologies can be characterized as either formal/top-level ontologies that describe the world in general or material/domain ontologies that describe specific domains [20]. Semantic Web ontologies (e.g., at the DAML library) are primarily domain ontologies. The development of domain ontologies was motivated by the need to develop systems that could reason with common sense knowledge of the real world [19]. More formally, a domain ontology is a catalog of the types of things that are assumed to exist in the domain of interest, D, from the perspective of a certain language, L, for the purpose of talking about that domain, D [45]. Conceptual modeling researchers have contributed extensively to the development and application of both formal and domain ontologies [2, 11, 13, 30, 50].

2.3 Information Retrieval

Methods from the information retrieval (IR) field [42] can inform the process of querying the Semantic Web and the role of domain ontologies. A core problem in IR is word-sense disambiguation: a word may have multiple meanings (homonymy), yet several words can have the same meaning (synonymy) [26, 36]. Resolving homonymy increases the relevance of the results returned (precision) by eliminating results of the wrong word-sense; resolving synonymy increases the proportion of relevant results in the collection returned (recall) by including terms that have the same meaning.

In IR, word-sense disambiguation involves two steps: 1. identifying the user's intended meaning of query terms, and 2. altering the query so that it achieves high precision and recall. In IR, the first step is usually achieved by automatically deducing a term's meaning from other terms in the query [1]. This is feasible because IR queries are typically long, e.g., 15 terms for short queries [9] and 50-85 for long queries [22]. On the Semantic Web, however, this appears infeasible. Most web queries are only two words long [46] and this is an insufficient length to identify context [9, 49]. Therefore, some user interaction will be required to accurately identify the intended sense of query-terms [1].

The second step (altering the query) is generally achieved in IR through a combination of:

- query constraints, such as requiring pages to include all query terms (possibly near each other) [22, 37].
- query expansion with 'local context,' in which additional terms are added to the query based on a subset of documents that the user identifies as relevant [37, 43]
- query expansion with 'global context,' in which additional terms are added to the query from thesauri, from terms in the document collection, or from past queries [9, 18, 29, 41, 49]

Of these methods, query constraints should be useful on the Semantic Web because they improve web search [38]. Query expansion with local context is less likely to be effective because reports indicate that web users rarely provide relevance feedback [8, 46]. Finally, query expansion with global context will remain relevant, but the sources used in global context analysis will be largely superseded by ontologies. For example, rather than use a thesaurus to add synonyms to a query so that relevant pages were not missed, a Semantic Web query could be left unexpanded and could simply rely on web pages referencing the terms on their pages to ontologies that defined each term and its synonyms. Recall, however, that ontologies will be of mixed quality [25]. Therefore, thesauri will likely remain important on the Semantic Web. The IR field suggests that two thesauri should be used (ideally in combination) [29, 34, 38, 49]: (1) general lexical thesauri that detail lexically related terms (e.g., synonyms), and (2) domain-specific thesauri that detail related terms in a specific domain. The preferred lexical thesaurus is WordNet, a comprehensive on-line catalog of English terms [15]. WordNet classifies the English language into synonym sets with underlying word senses (e.g., the noun “chair” has 4 word senses). WordNet has been found useful for traditional and web IR [38, 49]. It is difficult, however, to identify domain-specific thesauri for all domains on the web [18]. A solution is to use the domain ontology libraries on the Semantic Web. Stephens and Huhns [47] have shown that large ontology libraries can provide useful knowledge even in the presence of individual ontologies that are incomplete or inaccurate. Ontology libraries, therefore, have a dual role on the Semantic Web: 1) as a source of definitions of terms on specific web pages, and 2) as a source of semantics that can assist query expansion.

In summary, the IR field provides several insights into methods that will be required for querying the Semantic Web. The following insights, in particular, have influenced the development of the methodology presented in this paper:

- the need for user-interaction to identify the context of terms in short web queries,
- the continued relevance of query expansion using global context analysis,
- the need to use lexical and domain thesauri in combination, and
- the important role of large libraries of domain ontologies as sources of semantics.

3 Methodology for Retrieving Information from Semantic Web

Semantic Web languages and query schemes are still being developed [7, 10]. Nonetheless, the usefulness of Semantic Web ontologies for querying can be tested on the current web. Consider Berners-Lee’s et al. query: *Find Mom a specialist who can provide a series of bi-weekly physical therapy sessions*. The need for disambiguation is clear if the query is transformed into a short, more ambiguous query, more closely approximating queries on the web [46]: *Find doctors providing physical therapy*.

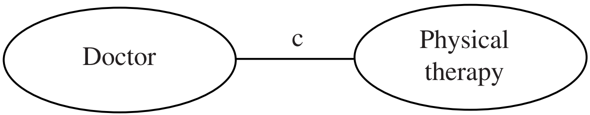
3.1 Overview of Methodology

A methodology for processing the query above is presented in Figure 1, above. Steps 1 to 3 identify the query context, as outlined below.

Identify Query Context →	**Build Knowledge Base** →	**Execute Query**
Step 1:	Step 4:	Step 7:
Identify noun phrases	Exclude incorrect sense	Construct Boolean query
Step 2:	Step 5:	Step 8:
Build synonym sets	Include hypernyms	Run query on search engine
Step 3:	Step 6:	Step 9:
Select word sense	Resolve inconsistencies	Retrieve and present results

Fig. 1. Methodology for Disambiguating, Augmenting, and Executing Query

Step 1 – Identify noun phrases. The methodology assumes that users enter their queries in natural language form, as users often have difficulty using Boolean logic or other query syntax [33, 46]. Nouns are identified using a modified form of the Qtag part-of-speech tagger [35]. Identifying phrases can significantly improve query precision [9, 33]. Thus, noun phrases are identified by querying each consecutive word-pair in WordNet. For example, "physical therapy" is a phrase in WordNet, so this step would find two noun phrases ('doctor' and 'physical therapy'). These noun phrases form the base nodes for expansion of the query. They are represented by a semantic network (see Figure 2). Initially, the terms are lined by a 'candidate' relationship, indicating that the terms have not yet been identified as relating in a lexical or domain-specific way. In the following stages, the semantic network will be augmented with three additional relationships: synonym (X is the same as Y), hypernym (X is a subclass of Y), and negation (X is not Y).



Key: c = candidate link

Fig. 2. Semantic Network of Query Terms after Step 1

Step 2 – Build synonym sets. To identify the different senses for each noun-phrase, synonym sets for each context are obtained from WordNet. For example, the term "doctor" has four senses in WordNet: 1. a medical practitioner, 2. a theologian, 3. a game played by children, and 4. an academician. "Physical therapy" has just one sense in WordNet. Each synonym set comprises one to many synonyms. These synonym sets for the query terms are incorporated into the knowledge base.

Step 3 – Select word sense. As the query does not contain enough terms to automatically deduce the word sense, user interaction is required [1, 49]. The user is presented with synsets for terms with multiple senses (e.g., "doctor") from which the user selects the most appropriate sense. Once the word-sense of terms has been identified, steps 4-6 build the knowledge base with additional terms to expand and constrain the query. The emphasis is on building a query that is biased towards precision, i.e., gives greater weighting to precision than recall [9, 22].

Step 4 – Exclude incorrect sense. To ensure that the results returned for a query are accurate, it is important to filter out those pages that contain incorrect senses of the term. Traditional query expansion does not include negative terms as filters [12, 18, 49]. Studies of web-query expansion have similarly not considered incorrect senses [28, 38]. Excluding incorrect senses is important on the web because of the vast number of results returned. Given that WordNet can return multiple synsets for each query term, incorrect senses can be inferred from the user's chosen word sense. For example, if a user selects the "medical practitioner" sense of 'Doctor,' then, a correct inference is that the user does not want pages associated with the other three senses of the term (theologians, children's games, or academics). Furthermore, because WordNet orders its synsets by estimated frequency of usage, terms can be chosen that are most likely to be successful in eliminating irrelevant results. We therefore use the following exclusion heuristic: *For each noun phrase, if a user identifies a synset as relevant, select the synonyms from the highest-ordered remaining synset, and include them in the knowledge base as negative knowledge* (see, e.g., Figure 3).

Step 5 – Including hypernyms. Past research has added hypernyms (or superclasses) to queries as a recall-enhancing technique, retrieving pages that contain either the search term (e.g., physical therapy) or its superclass (e.g., therapy) [18, 49]. Because precision is preferred over recall on the web [9, 22], our methodology uses hypernyms to increase the precision of queries by including them as mandatory terms. For example, rather than searching for *doctor OR "medical practitioner*," pages would be searched that contain *doctor AND "medical practitioner*." Of course, some 'doctor' pages that are relevant to the user may not include the hypernym "medical practitioner." Nevertheless, pages that contain 'doctor' and 'medical practitioner' are expected to be more likely to be consistent with the medical sense of the term 'doctor' than if the page only contained the 'doctor' term alone. Recall is, thus, sacrificed for precision. Following this approach, hypernyms are obtained from both WordNet and the DAML ontology library. WordNet provides the hypernyms for each synset automatically. The hypernyms from the DAML ontology library are obtained by querying the library for each noun phrase. The hypernyms from DAML and WordNet are then incorporated into the knowledge base. Table 1 shows the terms extracted from WordNet and the DAML ontology library from this step based upon the example. Figure 3 illustrates the expanded semantic network of terms after completing this step.

Step 6 – Resolve inconsistencies. The ability of query expansion to improve a query is dependent upon the quality of the terms added. Inconsistent terms could be added to the knowledge base when: (a) the synonym sets in WordNet are not orthogonal so a word-sense may be partially relevant but excluded by our methodology in step 4, (b) the DAML ontologies are of mixed quality so might contain inaccurate information [25], and (c) WordNet and the domain ontologies represent the contribution of many individuals who may have conflicting views of a domain. To identify inconsistencies, the methodology uses the following heuristic: *Check the hypernyms of the query terms (from DAML and WordNet) against the synonyms of the query term (from WordNet) that the user did not select as the desired word sense. Upon finding a match, ask the user if the term is desired. Adjust the knowledge base accordingly.*

Once the knowledge-base has been expanded, steps 7–9 are used to build and execute the query.

Table 1. Extraction of Hypernyms from WordNet and the DAML Ontology Library

Knowledge from WordNet		
Term	Word-sense (defined by synsets)	Hypernym (superclass)
Doctor	*Doc, Physician, MD, Dr, medico*	*Medical practitioner, medical man*
	Doctor of Church	*Theologian, Roman Catholic*
	Doctor	*Play*
	Dr	*Scholar, scholarly person, student*
Physical Therapy	*Physiotherapy, physiatrics*	*Therapy*
Knowledge from DAML ontology library		
Term	Hypernym (superclass)	
Doctor	Qualification, Medical care professional, Health professional	
Physical therapy	Rehabilitation, Medical practice	

Step 7 – Construct Boolean query. Following [22, 38], we construct the query using Boolean constraints to improve precision. Three heuristics are used:

(a) Synonym: *Automatically include the first synonym in the synset selected by the user and include it in the query with an OR, e.g., (query term OR synonym).*

(b) Hypernym: *Automatically include the first hypernym from WordNet for the user's selected word-sense. Allow the user to select up to one hypernym from those from the DAML library. Require the results to include either one or both hypernyms, e.g., query term AND (WordNet hypernym OR DAML hypernym).*

(c) Negation: *Automatically include the first synonym from the first synset in WordNet's list that was not selected by the user with a Boolean NOT, e.g., (query term NOT synonym).*

The rationale for each heuristic is to improve precision while minimizing user interaction. The synonym heuristic is used because the user may not have selected the most precise query term for that word sense. While this heuristic on its own could increase recall at the cost of precision [18], the use of a synonym in combination with a hypernym should improve precision. Because WordNet lists terms in estimated order of frequency of use, the first synonym is likely the best alternative for that term.

The hypernym heuristic is used to force pages to include the query term (or synonym) as well as the hypernym should increase the likelihood that the page contains the sense of the term the user desires. Following [34], WordNet and the DAML ontology are used in combination by allowing either of their hypernyms to be found. Because the DAML ontologies can be of mixed quality and because they do not provide hypernyms by word sense, user interaction is required to select the appropriate hypernym obtained from the domain ontologies.

Finally, the negation heuristic is used to filter out unwanted word senses. As WordNet can provide many synonyms in each synset and because terms are listed in estimated order of frequent usage, the first synonym from the first remaining synset is chosen as the most useful term for excluding pages.

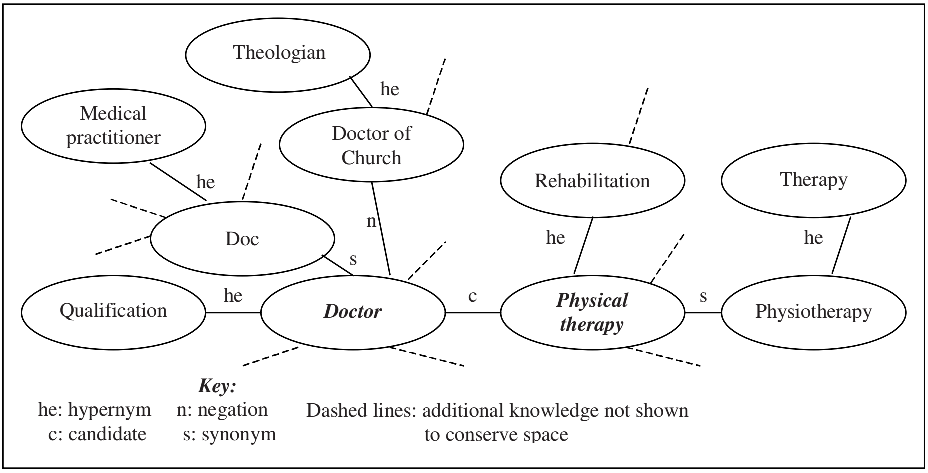


Fig. 3. Semantic Network of Query-Terms After Building Knowledge-Base

Applying these heuristics to the example, the following query is constructed: *(Doctor or Doc) and ("Medical practitioner" or Qualification) –"Doctor of church" and ("Physical therapy" or physiotherapy) and (therapy or "medical practice").*

Step 8 – Run query on search engine. Although a number of IR search engines have been developed, (e.g., for TREC), there is evidence that web search engines are more effective for web querying [44]. The methodology, therefore, submits the query to one or more web search engines (in their required syntax) for processing. The query construction heuristics are designed to work with most search engines. For example, while Altavista.com allows queries to use a NEAR constraint, this is unavailable on other search engines (e.g., Google.com, Alltheweb.com), so it is not used. Likewise, query expansion methodologies in IR can add up to 800 terms to the query with varying weights [41]. This approach is not used in our methodology because web search engines limit the number of query terms (e.g. Google has a limit of ten terms).

Step 9 – Retrieve and present results. In the final step, the results from the search engine (URLs and 'snippets' provided from the web pages) are retrieved and presented to the user. The user can either accept the query or revise the query to get more relevant results.

A pilot-test of the methodology indicated that the approach is feasible [5]. Queries using the above steps returned fewer results of equal or more relevance, with results dependent on the length and ambiguity of the query. A prototype was thus implemented to enable more detailed testing.

4 Implementation

A prototype called ISRA (Intelligent Semantic web Retrieval Agent) has been developed using J2EE technologies and informally tested [48]. ISRA uses the traditional client-server architecture as shown in Figure 4. The client is a basic web browser, through which the user specifies search queries in natural language. The server contains Java application code and the WordNet database. The prototype also provides an interface to several search engines including Google (www.google.com), Alltheweb (www.alltheweb.com) and AltaVista (www.altavista.com).

The prototype consists of three agents: a) Input-Output-Parser Agent, b) WordNet Agent, and c) Query Refinement and Execution Agent. These were implemented using a combination of jsp pages and servlets. The input-output-parser agent is responsible for capturing the user's input, parsing the natural language query, and returning results. The agent uses "QTAG", a *probabilistic parts-of-speech tagger* (available at http://web.bham.ac.uk/o.mason/software/tagger/index.html), to parse the user's input. It returns the part-of-speech for each word in the text. Based on the noun phrases (propositions) identified, an initial search query is created.

The WordNet Agent interfaces with the WordNet lexical database via JWordNet (a pure Java standalone object-oriented interface available at http://sourceforge.net/projects/jwn/). The prototype uses WordNet 1.6 (PC). For each noun phrase, the agent queries the database for different word senses and requests that the user select the most appropriate sense for the query. The agent extracts word senses, synonyms and hypernyms (superclasses) from the lexical database and forwards them to the query refinement agent to augment the initial query.

The Query Refinement and Execution (QRE) agent expands the initial query based on word senses, and synonyms obtained from WordNet. The refined query is then submitted to the search engine using appropriate syntax and constraints, and the results returned to the user. For example, the agent interacts with Google through its Web API service and adheres to the ten word limit for query length and displays ten hits at a time. Essentially, the QRE agent applies the steps shown in Figure 1 to augment the initial query. For example, it searches for phrases (word pairs) and includes them within double quotes, adds synonyms (from the WordNet synset) to the query based on the word sense selected by the user, and adds negative knowledge (terms) from the remaining word senses (e.g., would exclude the theologian sense of doctor if a query was for a medical doctor). For each term, the hypernym corresponding to the selected word sense is also retrieved from WordNet and DAML ontology and added to the query. The refined query is then sent to the search engine.

5 Testing

The effectiveness of the methodology was assessed by carrying out a laboratory study in which the results obtained using ISRA were compared to those obtained using the Google search engine alone. As the control group (Google) and the experimental group (Google plus methodology) used the same search engine, the experiment directly tests the benefit of the methodology and its heuristics.

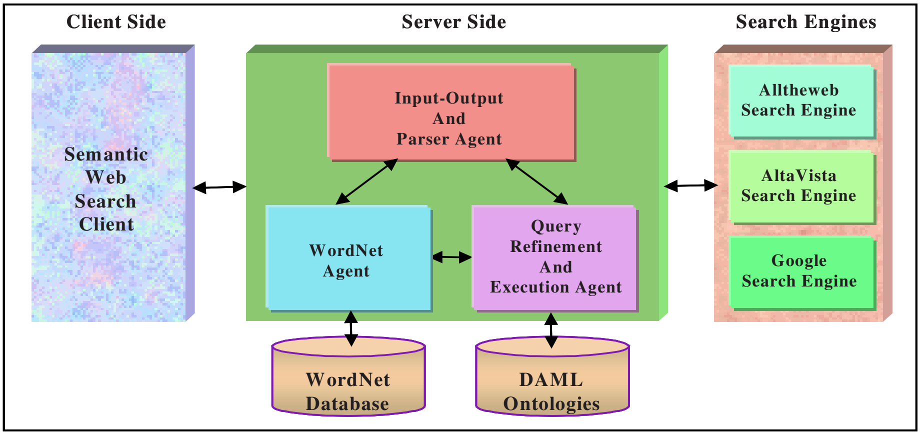


Fig. 4. ISRA Prototype Design

Forty-nine students from two universities participated voluntarily. All were experienced and frequent search engine users. Subjects were required to build their own queries and evaluate their own results since experimenters are unable to create as diverse a set of queries as users, can bias results, and cannot objectively determine whether a result is relevant to a user [17].

5.1 Dependent Variable and Hypothesis

Two measures of precision were used: the number of relevant pages in the first 10 and first 20 pages (Precision(10) and Precision(20)). These suit a web context because users do not view many results [4, 46]. Recall was not tested because it is less relevant and not strictly measurable on the web [12].

To test the flexibility of the methodology, the experiment tested the system's performance on two types of terms: ambiguous and clear. *Query term ambiguity* refers to the degree to which the query terms contain many word senses. Because the methodology aims to improve query precision, it should provide the most benefits when query terms have many word senses. Formally, the following hypothesis was tested:

Hypothesis: *ISRA will produce more relevant results than Google for queries with ambiguous terms.*

5.2 Query Sample

A large body of diverse user-developed queries was tested (per [4]). Some queries were expected to contain terms that did not exist in WordNet or the DAML library, e.g., 'instance' information such as product names. To alleviate this problem, and to test the hypothesis, the query sample was constructed as follows. First, all single-word classes in the DAML library were extracted. These were then pruned by excluding terms that (a) had no superclasses in the library, or (b) would be unknown to subjects (e.g., very specialized terms). 268 terms were obtained. These were divided into two groups based on their word-senses in WordNet. 131 terms (the clear group) had 0-2 word-senses (where 0 represents a domain-specific term not in WordNet). 137 terms (the ambiguous group) had 3 or more word-senses. In the experiment, subjects constructed four queries. Two queries could use any words. Two were constrained to a random list of 20 terms from DAML. Subjects were randomly assigned either 20 clear or unclear terms. Subjects used these to construct the two constrained queries.

5.3 Procedure

Each subject received instructions explaining the system's operation, how to construct queries, and how to grade web-page relevance. Subjects were first given a 10-minute training exercise to introduce them to the system and to practice developing a query and ranking results. Next, each participant developed his or her four queries. Each student's materials included the random list of clear or unclear terms for constructing their two constrained queries. The system provided the results from both ISRA and Google (in random order) and asked the subject for his or her ranking. Subjects were given three minutes to perform each query and four minutes to rank the first twenty pages returned for each query (two minutes to rank the results from ISRA and two minutes to rank the results from Google). A short time was given for assessment because this was considered to be typical web user behavior. Overall, the experiment took approximately 45 minutes.

5.4 Summary of Results

Table 2 summarizes the results. For each query, the methodology was expected to increase the number of relevant pages, decrease the number of irrelevant pages, and reduce the number of hits. Paired-sample t-tests identify the difference between groups. To assess the benefit of the heuristics, all subjects' queries were analyzed to determine if they invoked the query construction heuristics in section 3.1 (step 7) above. A more detailed analysis of each heuristic (e.g., to compare the contribution of terms from WordNet versus terms from DAML) is being examined in future research. The results for the experiment are:

All differences between the system and Google were in the expected direction

- For all queries, the methodology significantly reduced the number of pages returned
- For the full sample: the number of irrelevant pages in the top 10 was significantly reduced
- For the ambiguous queries: all results were significant and supported the hypothesis
- For the full sample, results were strong when heuristics were invoked. Using synonyms, hypernyms, and/or negative knowledge decreased the irrelevant pages. Using synonyms and/or negative knowledge increased the relevant pages.

Table 2. Summary of Results

Variable	Hypothesized Direction	All	Unclear	Syn	Hyp	Neg	SNH
R(10)	+	+.279	+.054*	+.091*	+.175	+.033**	+.003**
R(20)	+	+.371	+.014**	+.180	+.453	+.050*	+.006**
NR(10)	-	-.035**	-.003**	-.002**	-.007**	-.006**	-.000**
NR(20)	-	-.106	-.002**	-.011**	-.034**	-.007**	-.000**
Hits	-	-.000**	-.000**	-.000**	-.000**	-.000**	-.000**
N	NA	156	44	96	123	95	63

Key: Cell entries are p-values for paired sample t-tests (ISRA vs. Google).
** significant at $\alpha < 0.05$ one-tailed, * at $\alpha < 0.10$ one-tailed
R(10) = # relevant in top 10 , R(20) = # relevant in top 20, NR = # not relevant (in top 10 & 20), N = Sample size, All = full set of queries minus missing values, Unclear = subset of queries for the ambiguous group, Syn = subset of queries that invoked synonym heuristic, Hyp = subset of queries that invoked hypernym heuristic, Neg = subset of queries that invoked negation heuristic, SNH = subsets of queries that invoked synonym, hypernyms, and negation heuristics.

6 Conclusion

A significant body of research has emerged to investigate ways to facilitate the development of the Semantic Web. There have been few attempts, however, to develop methodologies for retrieving information from the Semantic Web. This paper presented a methodology for the development of an intelligent agent that makes effective use of lexicons and ontologies for processing queries on the Semantic Web. The methodology is based upon a desire to obtain good results from a query with minimal user intervention. The methodology builds upon research on natural language processing, knowledge-based systems, and information retrieval. Initial results show that the approach is beneficial. A prototype system was found to improve query results over a common search engine when (a) query terms were ambiguous, and/or (b) when the methodology's heuristics were invoked. Further work is needed to improve the scalability and customizability of the approach, and minimize user interaction.

References

1. Allan, J. and H. Raghavan. *Using Part-of-Speech Patterns to Reduce Query Ambiguity*. in *25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2002. Tampere, Finland: ACM Press, New York.
2. Bergholtz, M. and P. Johannesson. *Classifying the Semantics in Conceptual Modelling by Categorization of Roles*. NLDB'01. 2001. Madrid, Spain.
3. Berners-Lee, T., J. Hendler, and O. Lassila, *The Semantic Web*, in *Scientific American*. 2001. p. 1–19.
4. Buckley, C. and E.M. Voorhees. *Evaluating Evaluation Measure Stability*. in *SIGIR*. 2000. Athens, Greece: ACM.

5. Burton-Jones, A., S. Purao, and V.C. Storey. *Context-Aware Query Processing on the Semantic Web*. in *Proceedings of the 23rd International Conference on Information Systems*. 2002. Barcelona, Spain, Dec. 16–19.
6. CACM, *Special issue on ontology.* Communications of ACM, Feb. 2002. 45(2) p. 39–65.
7. Crow, L. and N. Shadbolt, *Extracting Focused Knowledge From the Semantic Web.* International Journal of Human-Computer Studies, 2001. 54: p. 155–184.
8. Cui, H., et al. *Probabilistic Query Expansion Using Query Logs*. in *Eleventh World Wide Web Conference (WWW 2002)*. 2002. Honolulu, Hawaii.
9. de Lima, E.F. and J.O. Pedersen. *Phrase Recognition and Expansion for Short, Precision-biased Queries based on a Query Log*. in *22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1999. Berkeley, CA.
10. Ding, Y., et al., *The Semantic Web: Yet Another Hip?* Data & Knowledge Engineering, 2002. 41: p. 205–227.
11. Dullea, J. and I.Y. Song. *A Taxonomy of Recursive Relationships and Their Structural Validity in ER Modeling*. Lecture Notes in Computer Science 1728. 1999. Paris, France.
12. Efthimiadis, E.N., *Interactive Query Expansion: A User-Based Evaluation in a Relevance Feedback Environment.* Jl. of American Society for Inf. Science, 2000. 51(11) p. 989–1003.
13. Embley, D.W., et al. *A Conceptual-Modeling Approach to Extracting Data from the Web*. in *17th International Conference on Conceptual Modeling, ER '98*. 1998. Singapore.
14. Farquhar, A., R. Fikes, and J. Rice. *The Ontolingua Server: a Tool for Collaborative Ontology Construction*. in *Tenth Knowledge Acquisition for Knowledge-Based Systems Workshop*. 1996. Banff, Canada.
15. Fellbaum, C. *WordNet: An Electronic Lexical Database*. 1998, MIT Press Cambridge, MA.
16. Fensel, D., et al., *OIL: An Ontology Infrastructure for the Semantic Web.* IEEE Intelligent Systems, 2001. March/April: p. 38–45.
17. Gordon, M. and P. Pathak, *Finding Information on the World Wide Web: The Retrieval Effectiveness of Search Engines.* Info. Processing & Management, 1999. 35: p. 141–180.
18. Greenberg, J., *Automatic Query Expansion via Lexical-Semantic Relationships.* Journal of the American Society for Information Science, 2001. 52(5): p. 402–415.
19. Gruber, T.R., *A Translation Approach to Portable Ontology Specifications.* Knowledge Acquisition, 1993. 5: p. 199–220.
20. Guarino, N. *Formal Ontology and Information Systems*. in *1st International Conference on Formal Ontology in Information Systems*. 1998. Trento, Italy: IOS Press.
21. Guha, R.V. and D.B. Lenat, *Enabling Agents to Work Together.* Communications of the ACM, 1994. 37(7): p. 127–142.
22. Hearst, M.A. *Improving Full-Text Precision on Short Queries using Simple Constraints*. in *SDAIR*. 1996. Las Vegas, NV.
23. Heflin, J., J. Hendler, and S. Luke, *SHOE: A Knowledge Representation Language for Internet Applications. Technical Report CS-TR-4078 (UMIACS TR-99-71).* 1999, Dept. of Computer Science, University of Maryland at College Park.
24. Hendler, J. and D.L. McGuinness, *The DARPA Agent Markup Language.* IEEE Intelligent Systems, 2000. 15(6): p. 67–73.
25. Hendler, J., *Agents and the Semantic Web.* IEEE Intelligent Systems, 2001. Mar p. 30–36.
26. Ide, N. and J. Veronis, *Introduction to the Special Issue on Word Sense Disambiguation: The State of the Art.* Computational Linguistics, 1998. 24(1): p. 1–40.
27. IEEE, *Special issue on the Semantic Web.* IEEE Intelligent Systems, Mar 2001: p. 32–79.
28. Jansen, B.J., *An Investigation Into the Use of Simple Queries on Web IR Systems.* Information Research: An Electronic Journal, 2000. 6(1): p. 1–13.
29. Jing, Y. and W.B. Croft. *An Association Thesaurus for Information Retrieval*. in *RIAO-94, 4th International Conference ''Recherche d'Information Assistee par Ordinateur''*. 1994.
30. Kedad, Z. and E. Metais. *Dealing with Semantic Heterogeneity During Data Integration. 18th Intl Conf on Conceptual Modeling, LNCS 1728*. 1999. Paris, France.

31. Kobayashi, M. and K. Takeda, *Information Retrieval on the Web.* ACM Computing Surveys, 2000. 32(2): p. 144–173.
32. Lawrence, S., *Context in Web Search.* IEEE Data Engg. Bulletin, 2000. 23(3) p. 25–32.
33. Lewis, D.D. and K. Spark Jones, *Natural Language Processing for Information Retrieval.* Communications of the ACM, 1996. 39(1): p. 92–101.
34. Mandala, R., T. Tokunaga, and H. Tanaka. *Combining Multiple Evidence from Different Types of Thesaurus for Query Expansion*. in *22nd Intl ACM SIGIR Conference on Research & Development in Information Retrieval*. 1999. Berkeley, CA.
35. Mason, O., *Qtag – a Portable POS Tagger*. 2003, online: http://web.bham.ac.uk/O.Mason/software/tagger/.
36. Miller, G.A., *Contextuality*, in *Mental Models in Cognitive Science*, J. Oakhill and A. Garnham, Editors. 1996, Psychology Press: East Sussex, UK. p. 1–18.
37. Mitra, M., A. Singhal, and C. Buckley. *Improving Automated Query Expansion*. in *21st Intl Conf on Research & Development on Information Retrieval*. 1998. Melbourne, Australia.
38. Moldovan, D.L. and R. Mihalcea, *Improving the Search on the Internet by using WordNet and Lexical Operators.* IEEE Internet Computing, 2000. 4(1): p. 34–43.
39. Mylopoulos, J., *Information Modeling in the Time of the Revolution.* Information Systems, 1998. 23(34): p. 127–155.
40. Noy, N.F. and C.D. Hafner, *The State of the Art in Ontology Design: A Survey and Comparative Review.* AI Magazine, 1997. 18(3/Fall): p. 53–74.
41. Qiu, Y. and H.-P. Frei. *Concept Based Query Expansion*. in *16th Annual Intl ACM SIGIR Conference on Research and Development in Information Retrieval*. 1993. Pittsburgh, PA.
42. Raghavan, P. *Information Retrieval Algorithms: A Survey*, *Eighth Annual ACM-SIAM Symp on Discrete Algorithms*. 1997. New Orleans, Louisiana: ACM Press, New York, NY.
43. Salton, G. and C. Buckley, *Improving Retrieval Performance by Relevance Feedback.* Journal of the American Society for Information Science, 1990. 41(4): p. 288–297.
44. Singhal, A. and M. Kaszkiel. *A Case Study in Web Search Using TREC Algorithms*. in *10th International World Wide Web Conference*. 2001. Hong Kong.
45. Sowa, J.F., *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. 2000: Brooks Cole Publishing Co.
46. Spink, A., et al., *Searching the Web: The Public and Their Queries.* Journal of the American Society for Information Science, 2001. 52(3): p. 226–234.
47. Stephens, L.M. and M.N. Huhns, *Consensus Ontologies: Reconciling the Semantics of Web Pages and Agents.* IEEE Internet Computing, 2001. September–October: p. 92–95.
48. Sugumaran, V., A. Burton-Jones, and V.C. Storey. *A Multi-Agent Prototype for Intelligent Query Processing on the Semantic Web*. in *Proceedings of the 12th Annual Workshop on Information Technology and Systems (WITS)*. 2002. Barcelona, Spain, Dec. 14–15.
49. Voorhees, E.M. *Query Expansion Using Lexical-Semantic Relations*. in *17th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval*. 1994. Dublin, Ireland.
50. Weber, R., *Ontological Foundations of Information Systems*. Coopers and Lybrand Accounting Research Methodology, Monograph No. 4. 1997, Melbourne: Coopers & Lybrand and Accounting Association of Australia and New Zealand. 212.